(12) UK Patent Application (19) GB (11) 2 356 106 (13) A

(43) Date of A Publication 09.05.2001

(21) Application No 0016661.1

(22) Date of Filing 06.07.2000

(30) Priority Data
(31) 60142633 (32) 06.07.1999 (33) US

(71) Applicant(s)
AT & T Laboratories Cambridge Limited
(Incorporated in the United Kingdom)
24a Trumpington Street, CAMBRIDGE, CB2 1QA, United Kingdom

(72) Inventor(s)
James Quentin Stafford-Fraser
Andrew Harter
Tristan John Richardson
Nicholas John Hollinghurst

(74) Agent and/or Address for Service
Marks & Clerk
4220 Nash Court, Oxford Business Park South, OXFORD, OX4 2RU, United Kingdom

(51) INT CL[7]
H04L 29/00

(52) UK CL (Edition S)
H4K KFH

(56) Documents Cited
GB 2332126 A GB 2321732 A EP 0903903 A2
EP 0355697 A2 WO 97/42728 A2
Taniguchi K Internet Video etc IEICE Trans. Comm. Vol.E79B No. 8 Aug.96 pp 1068-1075 ISSN 0916-8516
Libman R E The Interactive Video etc AT&T Tech J. Vol. 74 No 5 Sep/Oct 95 pp 92-105 ISSN 8756-2324

(58) Field of Search
UK CL (Edition S) H4K KFH KOD3 KOD4 KOD8
INT CL[7] H04L 29/00 , H04Q 11/04
Online:WPI,EPODOC,JAPIO,INSPEC

(54) Abstract Title
Multimedia client-server system

(57) A communication system and method comprises endpoint devices 10, 11, each of which has one or more audio transducers and a touch screen. The devices 10, 11 are connected by a non-dedicated network 12 providing non-dedicated communication paths to servers 14, 15. An application 16, 17 is resident in each of the servers 14, 15 and has the ability to affect the image displayed on at least part of the screen. The server 14, 15 performs signaling for controlling an audio connection between the devices 10, 11. The touch screen is interactive and is able to initiate the audio connection. The application 16, 17 allows the screen or each screen of devices 10, 11 participating in the audio connection to display the path of consecutive measured positions of a pointer on the screen from one or more of the connected devices 10, 11. The screen is able to display an image supplied by a remote server or other apparatus after the audio connection has been initiated. This can be used for telephony, etc.

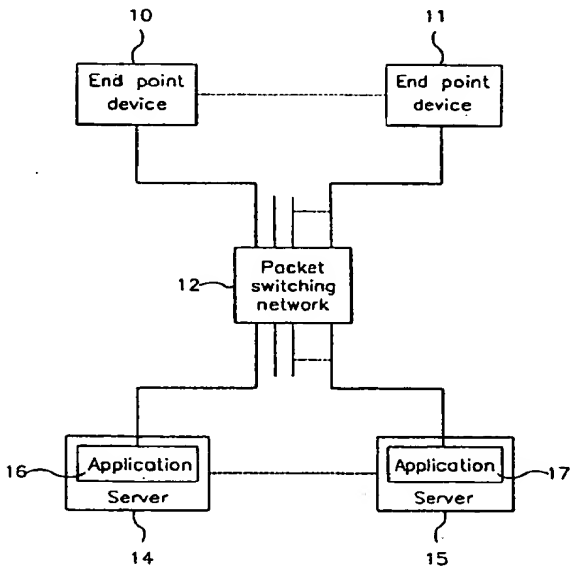


FIG 1

GB 2 356 106 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

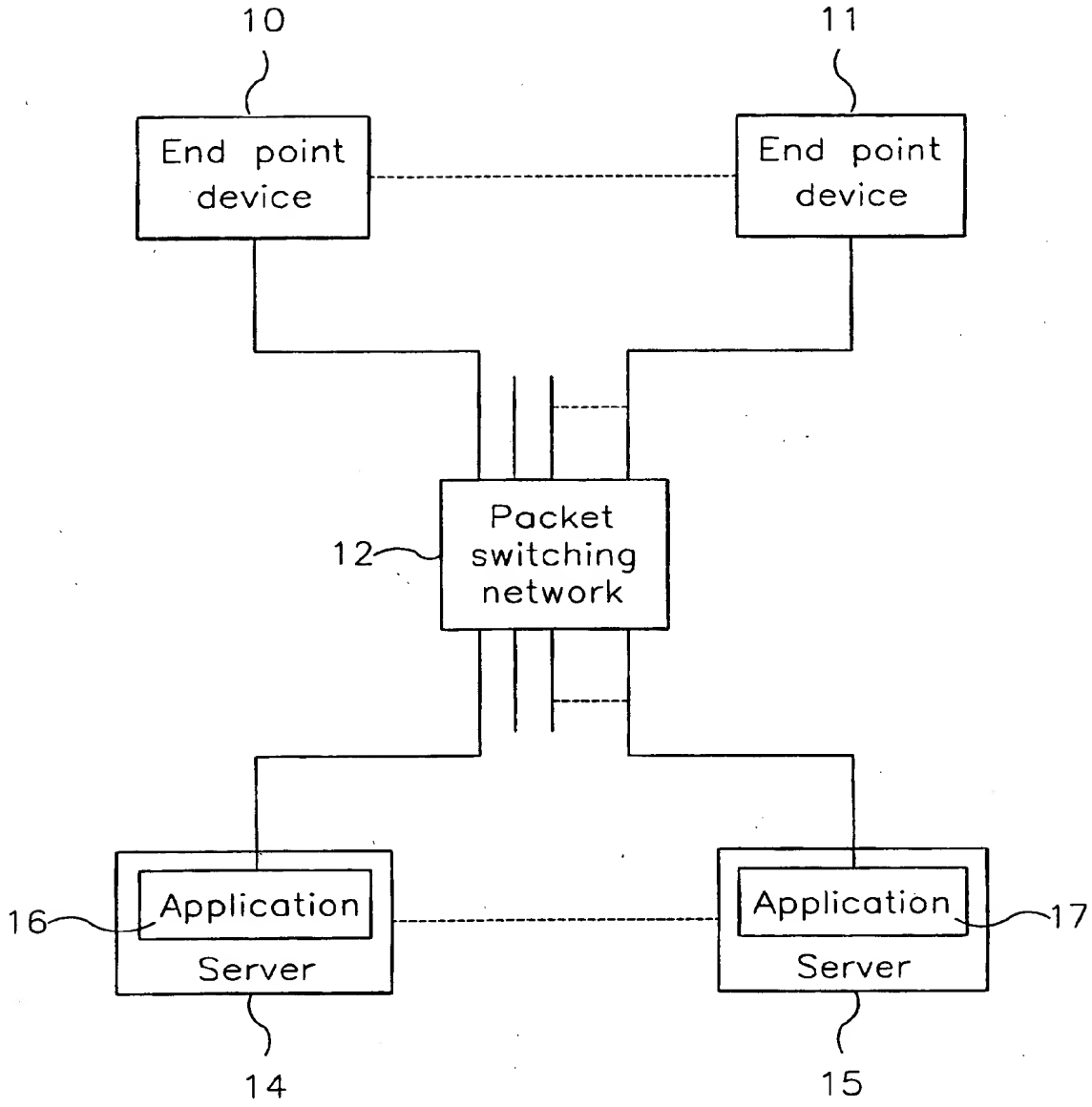

10
11
End point device
End point device
Packet switching network
12
Application
Server
16
Application
Server
17
14
15

FIG 1

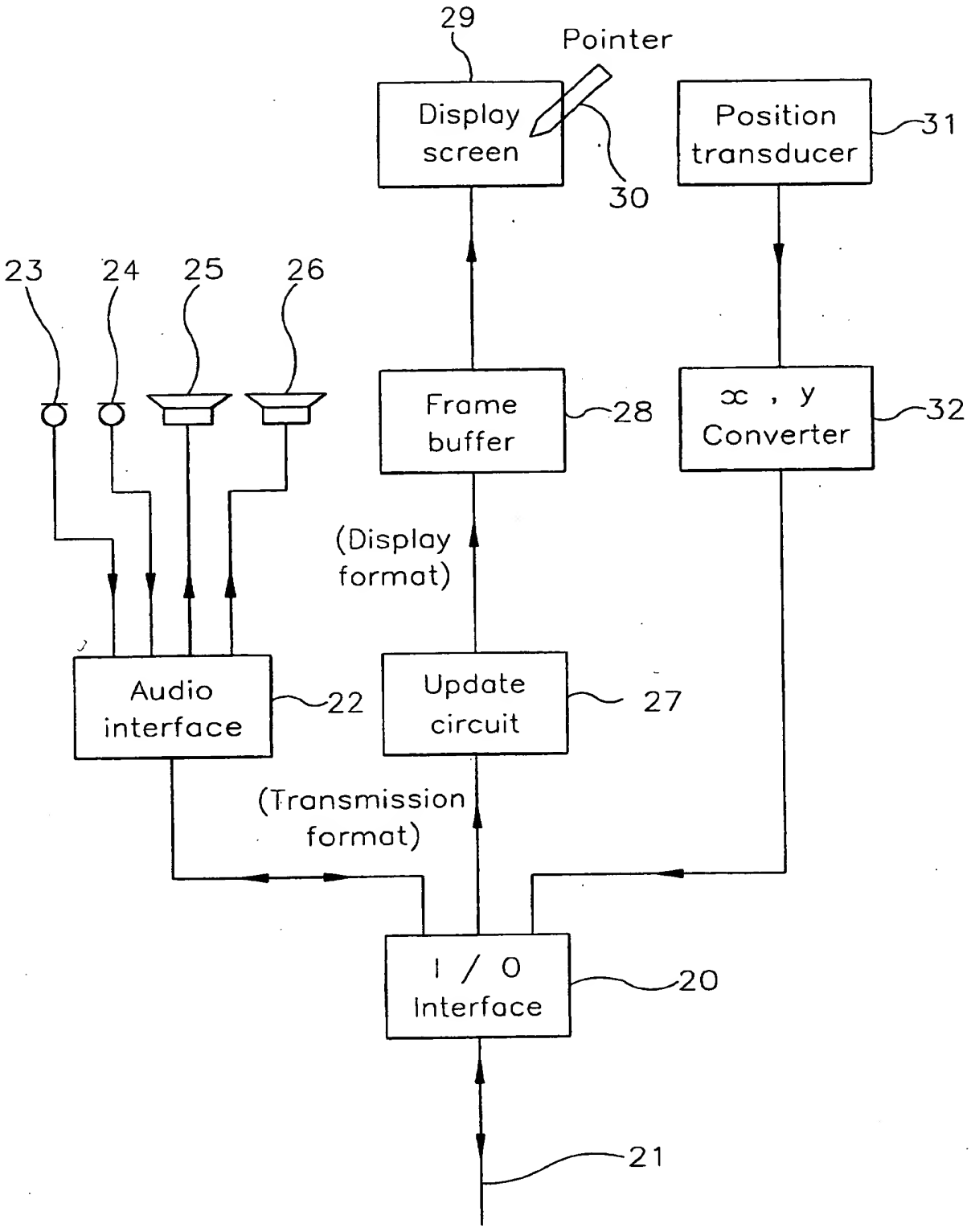

29
Pointer
Display screen
30
Position transducer
31
23
24
25
26
Frame buffer
28
x , y Converter
32
(Display format)
Audio interface
22
Update circuit
27
(Transmission format)
I / O Interface
20
21

FIG 2

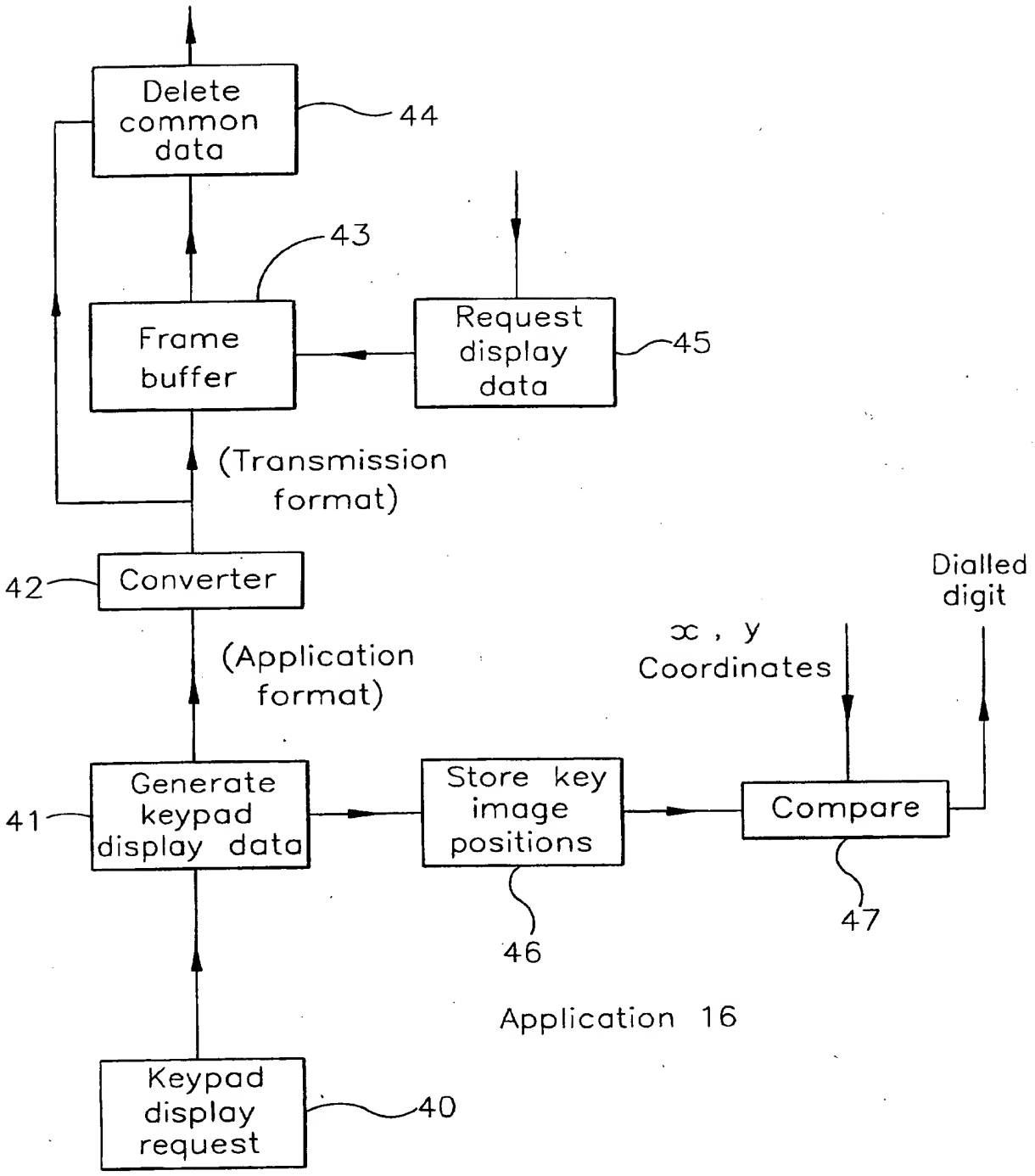

Delete common data
44
43
Frame buffer
Request display data
45
(Transmission format)
42
Converter
(Application format)
Dialled digit
x , y Coordinates
41
Generate keypad display data
Store key image positions
Compare
46
47
Application 16
Keypad display request
40

FIG 3

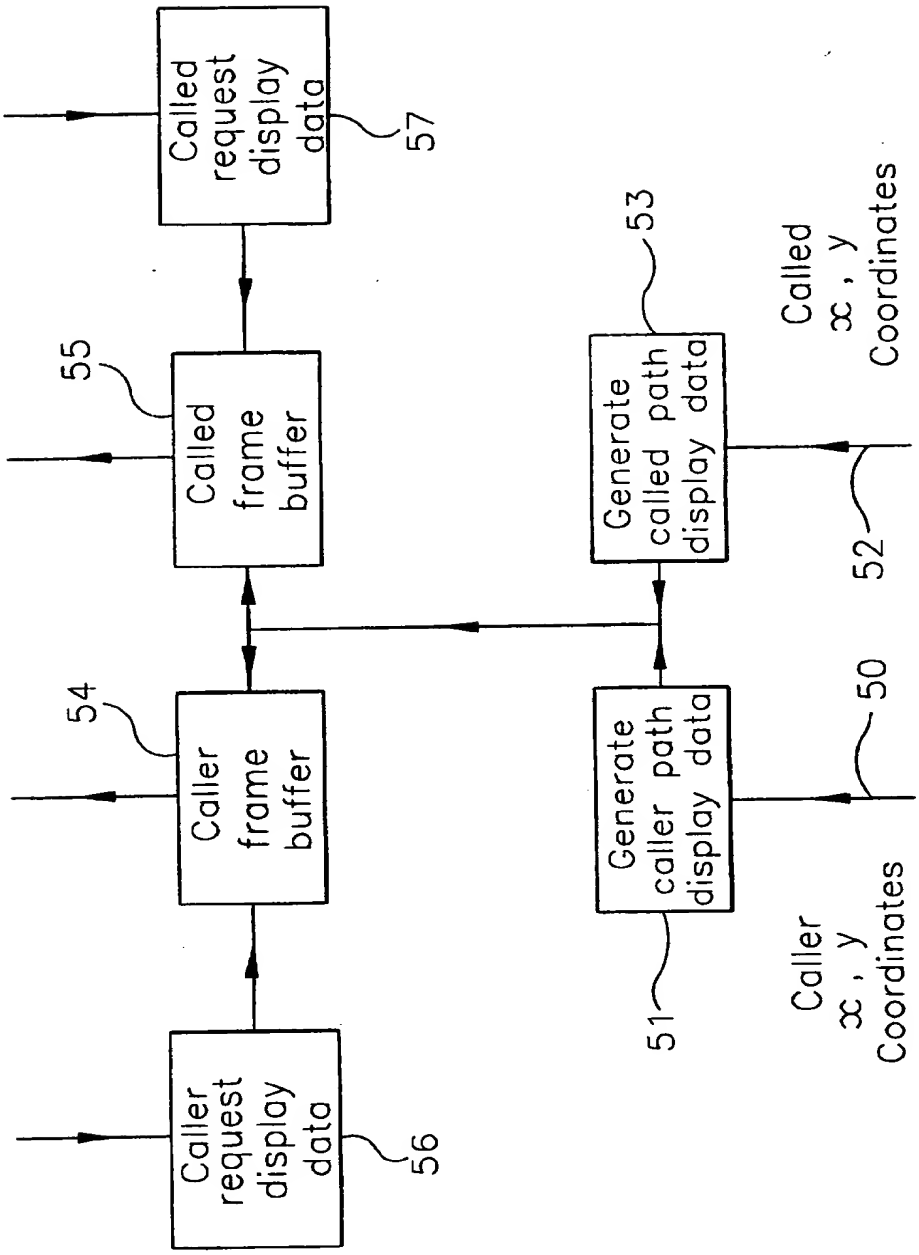

Called request display data
57
Called frame buffer
55
Caller frame buffer
54
Caller request display data
56
Generate called path display data
53
Called x , y Coordinates
52
Generate caller path display data
50
51
Caller x , y Coordinates

FIG 4

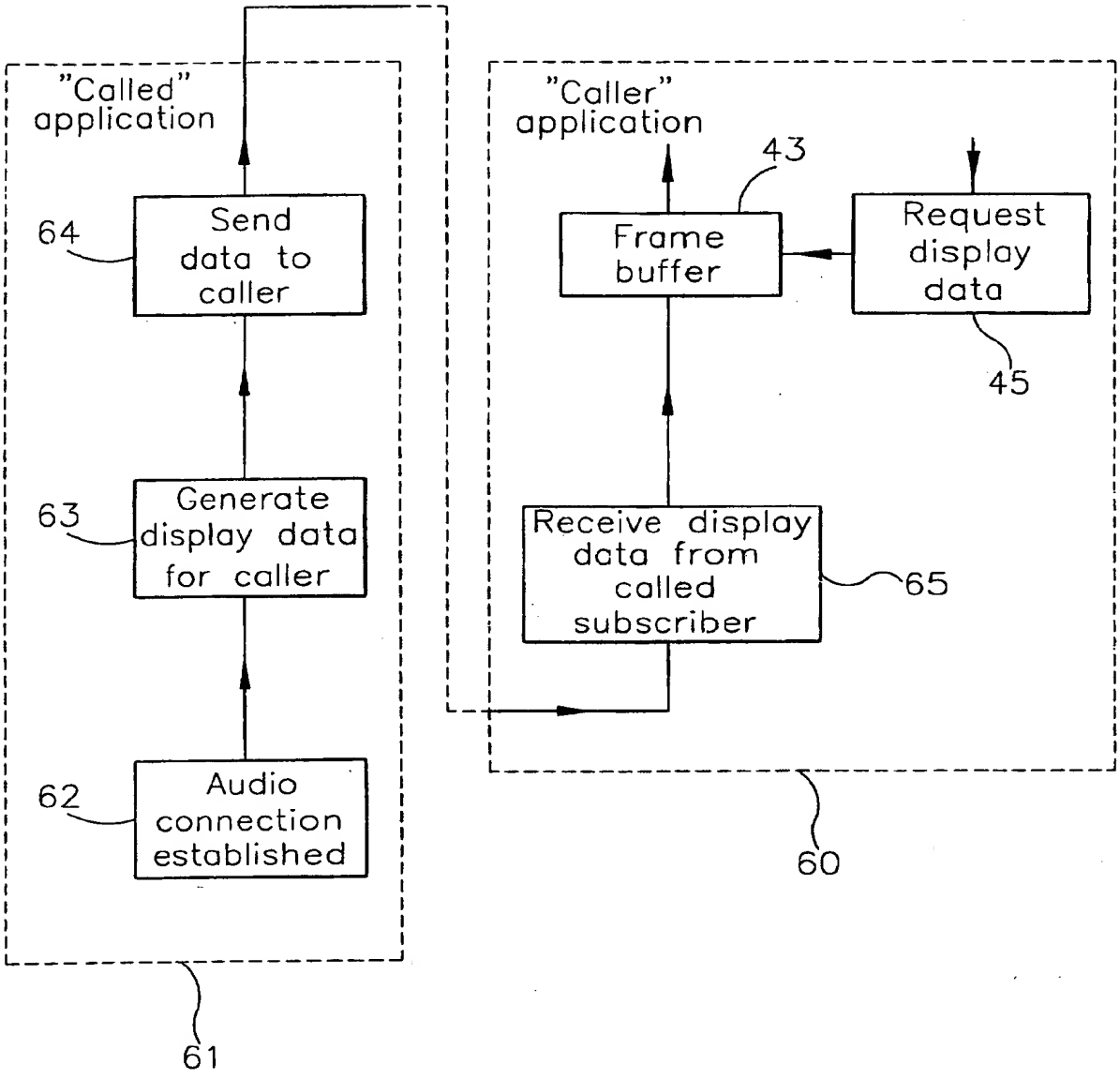
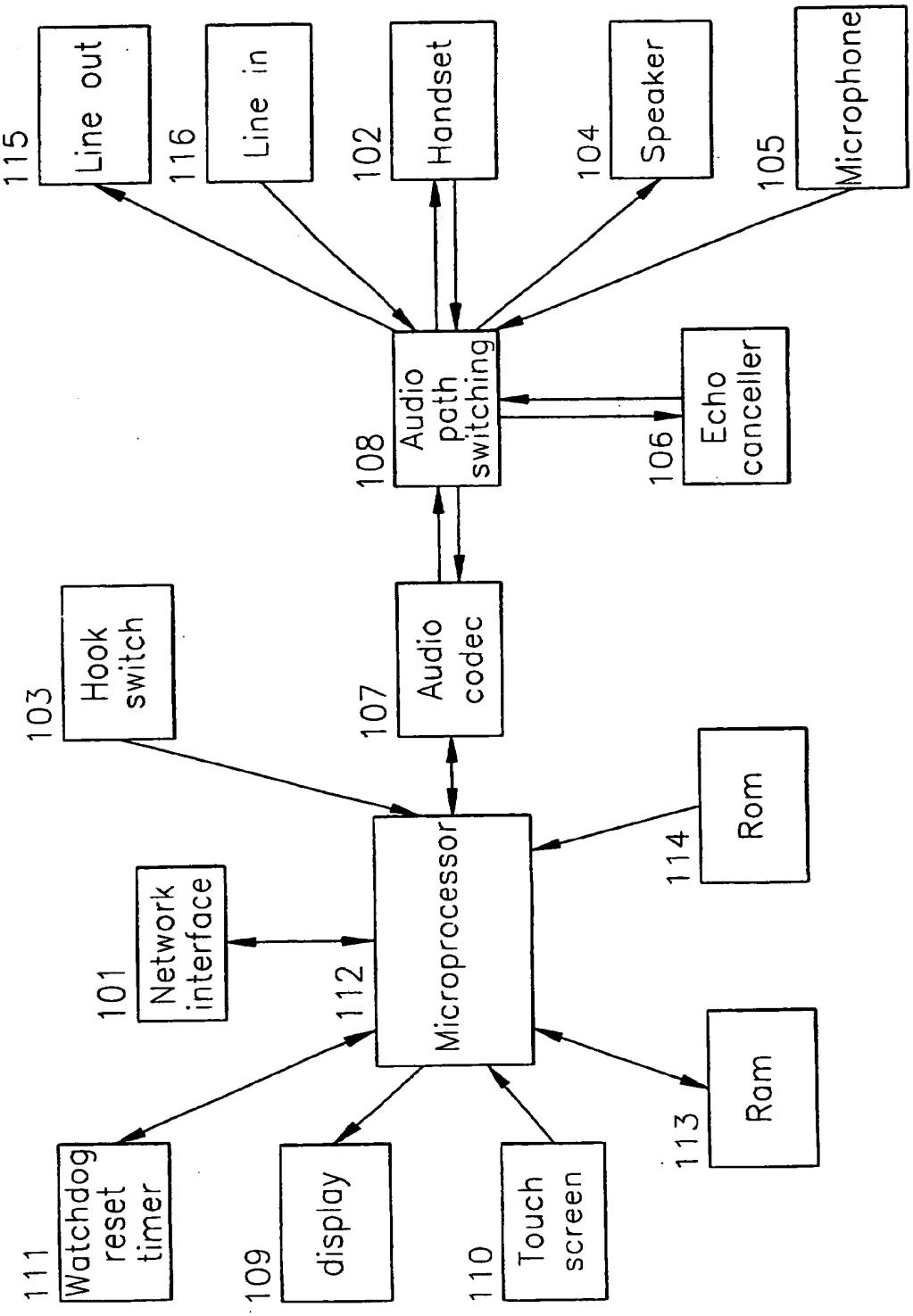
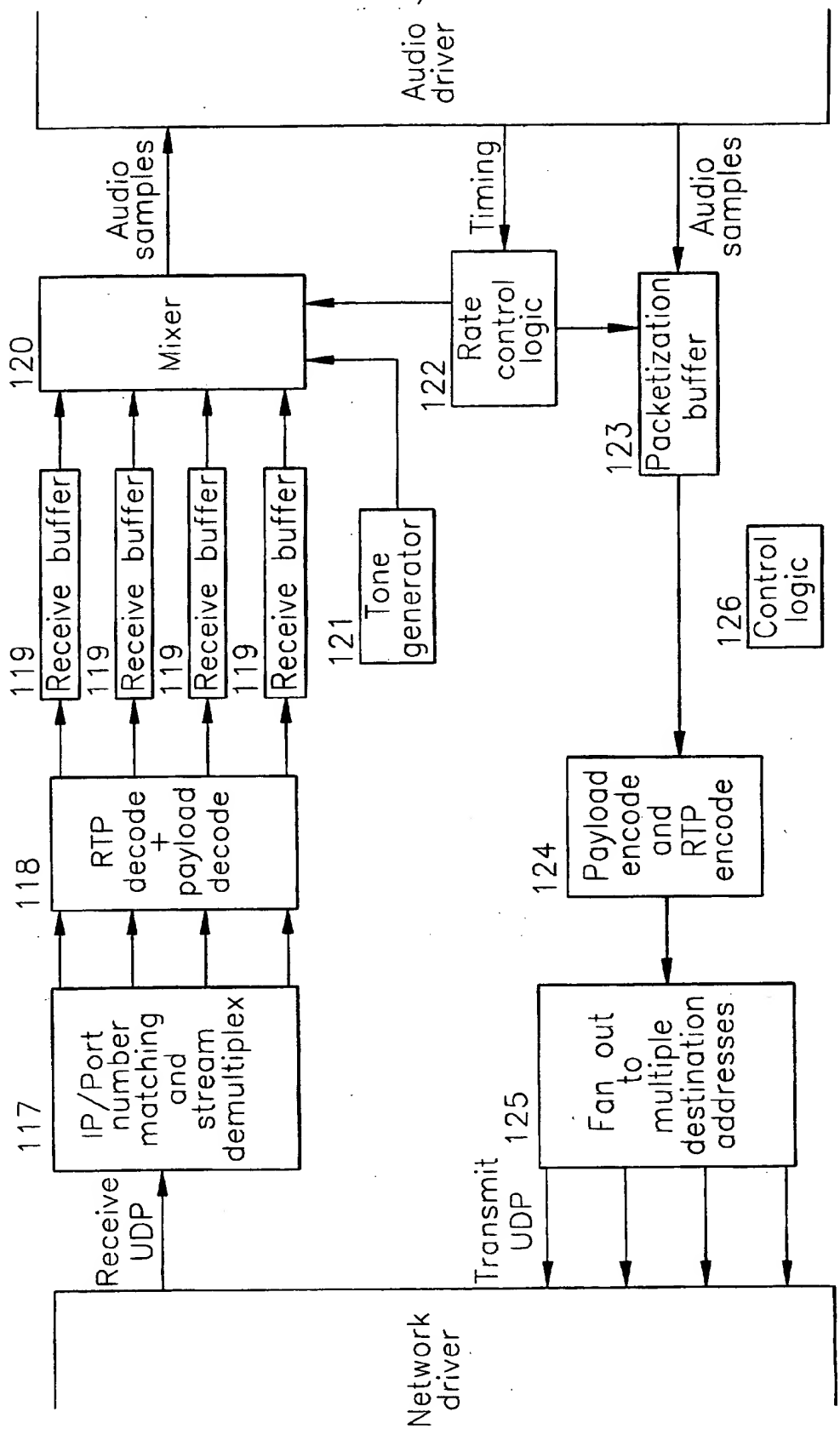

"Called" application
64
Send data to caller
63
Generate display data for caller
62
Audio connection established
61
"Caller" application
43
Frame buffer
Request display data
45
Receive display data from called subscriber
65
60

FIG 5



Device hardware overview

FIG 6



Audio transceiver functional blocks

FIG 7

A Thin Multimedia Communication Device and Method

Claim of Priority

This application claims priority from United States Provisional Application 60/142,633, filed in the United States Patent and Trademark Office on July 6, 1999, attorney docket 1999-0382.

Related Applications

This application is related to the three other identically titled applications filed on the same day as this application, naming the same inventors and commonly assigned as of the filing, Serial Nos. to be enumerated here upon receipt.

Incorporation by Reference

This application incorporates by reference United States Provisional Application 60/142,633, filed in the United States Patent and Trademark Office on July 6, 1999, attorney docket 1999-0382, with the same effect as if it appeared in this application verbatim, including all drawings.

Background of the Invention

Field of the Invention

This invention relates to audio and visual communication methods and devices. More particularly, the invention relates to the transmission of visual and audio information to a relatively dumb client, often called a thin client.

Description of the Art

In the early days of computing mainframe devices dominated the field. Computing tasks had to be programmed into these machines by direct access, such as punched cards. As time

progressed such machines became remotely accessible through otherwise "dumb" terminals such as teletypes. More powerful computer chips led to the next step in the development of computers - the personal computer or PC. This device had sufficient power to solve many problems and run many applications such as word processors, spreadsheets and databases. The PC had a keyboard for data entry and dumb terminals were no longer used. Essentially all devices, such as keyboards, were directly connected to a computer dedicated to that device. However, as transmission lines, such as optical fiber, are able to handle ever-higher bandwidth, the pendulum is swinging back, and relatively dumb terminals, sometimes known as "thin clients", are coming back into vogue. Such thin clients are connected to powerful computers or "servers" on which most of the necessary computation occurs.

VNC is one form of thin client architecture. VNC is a platform independent protocol that is used to transmit sections of a frame buffer from a server to relatively dumb client software, usually associated with a display device. A typical use of VNC involves display of a work-station desk-top on a remote machine which may be of a different architecture than the work station. The VNC protocol does not support transmission of audio other than a simple beep command.

Telephones, including IP telephones with displays, usually perform at least some signaling functions.

Microsoft NetMeeting is a toolkit that enables sharing of applications between machines. Microsoft NetMeeting does not involve thin devices but does include audio and other media transmission.

Teleporting is a system that employs a proxy to enable the X Windows protocol which uses end points that are not stateless, to cope with disconnection and reconnection of the endpoints. X Windows does not support transmission of audio other than a simple beep.

Medusa is a networked multimedia architecture in which computers perform signaling for stateless multimedia devices that are networked. In the Medusa model, audio and visual display are not contained in a single device.

Summary of the Invention

In one embodiment, this invention is a method and structure involving a stateless, relatively thin, communication device that has, in addition to audio capability, a display screen. (For convenience, we refer to this device as a broadband phone.) In accordance with the relatively thin nature of the device, a server process running on a server machine sends appropriate pixel information to illuminate the display screen. This information may result, for example, in the appearance of a telephone-dialing pad on the display screen. When the user touches numbers on the displayed telephone pad, the only information transmitted to the server is the location on the screen where the contacts were made. The server translates the location information into the number displayed and performs the appropriate signaling over the network. Accordingly, and consistent with the relatively thin nature of the device, signaling, such as that associated with establishing a communication link to another end-user device, is performed at the server rather than at the communication device. Since the display is generated by pixel information sent from the server, if the display device is disconnected, the server can simply regenerate the screen upon reconnection of the display device – a characteristic of stateless devices. Other embodiments of the invention include, for example, the display of a scribble pad on the screen and the ability to view in real time the creation of notes by any one of a plurality of parties that are connected both acoustically and visually. In yet another embodiment, a called party can cause information to be displayed on the screen – for example, the display of a menu when a fast food establishment is called. In other embodiments, appropriate web pages can be displayed on the screen via access to the Internet. In all of these embodiments, areas of the screen can be touched to obtain yet further information or actions.

According to a first aspect of the invention, there is provided a system as defined in the appended claim 1. Other aspects and embodiments of the invention are defined in the other appended claims.

The term "server" as used herein is defined to mean a process or set of processes resident on one or more data processors, such as computers. The term "non-dedicated communication path" as used herein is defined to mean a communication path through a network which is not a dedicated point-to-point path. An example of a non-dedicated path is a path through a routable network where information is routed between points of the network as a result of an addressing scheme, such as a packet switching network. The term "audio transducer" as used herein is defined to mean a device which converts audio frequency acoustic energy, such as speech, to corresponding electrical signals and/or vice versa. The term "application" as used herein is defined to mean a collection of user-interface interactions and systems effects which has a semantic theme.

Brief Description of the Drawings

Figure 1 is a block schematic diagram of a communication system constituting an embodiment of the invention;

Figure 2 is a block schematic functional diagram of an endpoint device of the system of Figure 1;

Figure 3 is a functional diagram illustrating the operation of a first server application of the system of Figure 1;

Figure 4 is a functional diagram illustrating the operation of a second server application of the system of Figure 1;

Figure 5 is a functional diagram illustrating the operation of a third server application of the system of Figure 1;

Figure 6 is a block schematic diagram illustrating in more detail an endpoint device of the system of Figure 1; and

Figure 7 is a block schematic functional diagram of an audio transceiver of the system of Figure 1.

Detailed Description of the Drawings

Figure 1 illustrates a communication system which constitutes an embodiment of the invention. The system comprises a plurality of endpoint devices, only two of which are illustrated at 10 and 11. The endpoint devices are in the form of multimedia communication devices and provide telephone services to the subscribers.

The endpoint devices are connected to a network 12 which is exemplified in Figure 1 by a packet switching network.

The system further comprises a plurality of servers, only two of which are shown at 14 and 15 in Figure 1. Each of the servers 14, 15 has residing therein one or more applications, only one 16, 17 of which is illustrated for simplicity. Each of the servers 14, 15 is associated with a respective endpoint device 10, 11 and performs signaling for controlling an audio connection which may be set up between any two or more of the endpoint devices 10, 11, for example to provide conventional telephone communication between two or more of the devices. Each of the applications 16, 17 affects the image which appears on at least part of a display screen of the respective endpoint device 10, 11.

Each of the endpoint devices is connected to its respective server by the packet switching network 12, which constitutes a non-dedicated communication path between the endpoint device and the corresponding server. The communication path is non-dedicated in the sense that it is not a dedicated point-to-point path. Instead, the network is routable such that information is routed between points of the network as a result of an addressing scheme, a packet switching network being a typical example of such a network.

The packet switching network 12 is capable of establishing connections between two endpoint devices, for example in the case of a typical telephonic connection, or more devices, for example in the case of a so-called "conference call". Connections may be established between endpoint devices which are physically connected to the same network.

Figure 2 illustrates an endpoint device comprising an input/output interface 20 which provides interfacing between, on a first side, the various remaining parts of the device and, on a second side, a non-dedicated communication path 21 in the form of a single channel which carries audio and non-audio data. The interface 20 is connected to an audio interface 22 which supports, for example, two acousto-electric transducers such as microphones 23 and 24 and two electro-acoustic transducers such as a loudspeaker 25 and an earphone 26. The interface circuit 20 is also connected to an update circuit 27 which in turn is connected to a frame buffer 28 for a display screen 29. The updated circuit 27 receives image data for updating the image on the screen 29 from the interface 20 and converts this from a transmission format to a display format. For example, the data may be encoded for transmission by data compression techniques so as to reduce the traffic over the network 12, in which case the circuit 27 decompresses the image data and supplies it in a format which is suitable for reading directly into the buffer 28.

The screen 29 is in the form of a touch screen or similar device and is illustrated in Figure 2 as having a pointer 30, which may, for example, be a stylus or similar device but might also be

the finger of a subscriber. The touch screen is illustrated as comprising a position transducer 31 which determines the position of a tip of the pointer 30 adjacent the display screen 29 in relation to the display screen. The signals provided by the transducer 31 are converted in a converter 32, for examples to signals representing the Cartesian coordinates x, y of the position of the pointer 30 relative to the screen 29.

Figure 3 illustrates a typical example of an application 16, for example resident in the server 14 and supporting the endpoint device 10. The application 16 supplies image data via the network 12 to the device 10 and causes the server 14 to perform signaling for controlling an audio connection between the device 10 and another such device connected to the network 12. The application 16 comprises an element 40 which responds to requests for generating a keypad display on the screen 29 of the device 10.

The element 40 actuates an element 41 which generates image data, in an application format, for producing an image on the display screen 29 of a keypad. In one example, the keypad image has the appearance of a conventional numeric keypad with numeric keys and other keys normally associated with a conventional telephone device. As an alternative or an addition, the element 41 may generate image data representing the names or pictures of subscribers of the communication system.

The display data in the application format is supplied to a converter 42 which converts the data to a transmission format. For example, this conversion may include converting to a format representing rectangular blocks of pixels and associated coordinates for locating the pixels at the appropriate place on the display screen 29. Although the converter 42 is illustrated as immediately following the element 41, it may be located elsewhere in the functional arrangement of the application 16.

The data from the converter are supplied to a frame buffer 43, which buffers the image data for transmission to the endpoint device 10. The output of the buffer 43 is supplied to an element 44 which checks items of image data for transmission to the device 10 and deletes common image data. In particular, whenever data are sent to the device 10 from the buffer 43, the element 44 detects when subsequent items of image data are intended for the same pixels on the screen 49. The element 44 ensures that only the most recent image data for each pixel is actually transmitted to the device 10 by deleting common pixel data from all earlier items.

The buffer 43 responds to requests 45 from the endpoint device 10 for display data to be transmitted thereto. Thus, the application 16 waits for a request from the device 10 indicating that the device is ready to receive fresh image data. Items of image data ready for transmission are stored in the buffer 43 until such a request is received, at which time the items are transmitted to the device 10. This arrangement ensures that image data are supplied in an efficient manner to the device 10 and in such a way that no image data are lost.

The application stores at 46 the positions of the images of the control keys on the screen 29 generated by the element 41. These positions are supplied to a comparator 47, which receives the position of the pointer 30 relative to the screen 29 in the form of x, y coordinates as determined by the transducer 31 and converted by the converter 32 of the device 10. The comparator 47 compares the position of the pointer with the stored positions of the key images and, whenever the pointer 30 is determined to be pointing to one of the key images on the screen 29, the comparator 47 provides the appropriate response. For example, as illustrated in Figure 3 where the image displayed on the screen 29 is of a numeric keypad, whenever the pointer 30 is determined to be pointing at one of the numeric keys, the comparator 47 supplies a signal representing a "dialed digit" corresponding to the key. For example, the dialed digit signal is supplied to the PSTN forming part of the network 12 and is used in the process of establishing an audio connection between the device 10 and another device connected to the network 12. Where the screen 29 displays names and/or images of subscribers to the communication system,

the comparator 47 may supply all of the connection signaling for connecting the device 10 to another endpoint device associated with a selected subscriber when the pointer 30 points to the appropriate name or image.

Figure 4 illustrates in simplified form another process which permits several subscribers (two in the example illustrated) to use the display screens 29 as "scribble pads" such that both subscribers can draw on a common scribble pad, for example by means of the pointers 30, and the resulting paths traced by the pointers are visible on the screens 29 of the endpoint devices of both subscribers. The application illustrated in Figure 4 is distributed between the servers associated with the endpoint devices of the two (or more) subscribers.

The caller x, y coordinates of the position of the pointer 30 relative to the screen 29 of the subscriber who requested the audio connection are received from the caller endpoint device at 50 and the display data representing the path traced by the caller pointer 30 on the screen 29 are generated at 51. Similarly, the called x, y coordinates are received from the called subscriber at 52 and are used at 53 to generate the display data representing the path traced by the pointer 30 on the screen 29 of the called subscriber. The caller and called display data are merged for updating the screens of the caller and the called subscriber. The merged display data are supplied to a caller frame buffer 54 and a called frame buffer 55. Thus, although both the caller and the called subscriber ultimately view the same image on their screens 29, the provision of the separate buffers 54 and 55 allows the respective endpoint devices to receive fresh image data when they are individually ready. For this purpose, the elements 56 and 57 receive the requests for fresh display data to be transmitted to the caller and the called subscriber respectively, and control their respective buffers 54, 55 in the same way as illustrated at 45 in Figure 3.

Figure 5 illustrates in simplified form a caller application 60, for example running on the server 14 of the endpoint device 10 which is requesting the establishment of an audio connection, and a "called" application 61, for example running of the server 15 of the device 11 to which the

audio connection is requested. This arrangement illustrates how display data may be automatically sent to the display screen 29 of a caller in response to the successful establishment of an audio connection, although the data may be sent at any time after an audio connection is initiated.

In the called application 61, the successful establishment of an audio connection is detected at 62 and, as a result of such detection, display data for sending to the caller is generated at 63. The display data may be in any appropriate format and, at 64, are sent by the server 15 to the server 14.

In the caller application 60, the data from the called application are received at 65 and are supplied to the buffer 43, which is controlled by the element 45 as described hereinbefore and as illustrated in Figure 3. Thus, as a result of establishing an audio connection between a caller and a called subscriber, the called subscriber can automatically send display data for displaying on the screen 29 of the caller. For example, the displayed images may represent a menu illustrating various items which are available for purchase by the caller. The caller may select one or more of the items by using the pointer 30 to point to the image of the or each selected item. This may be detected by an application of the type illustrated in Figure 3 and the selection may be signaled to the called application in order to initiate or make a purchase of the selected item.

Specific Embodiment

A specific embodiment of the invention includes a telephone-like appliance incorporating audio input and output devices, an LCD (Liquid Crystal Display) touchscreen, a network connection and a server.

The telephone like appliance includes a 'thin-client' graphics viewer that receives updates to its screen display from a network server, and sends back to the server information relating to finger/pen-presses on the touchscreen. A wide variety of applications and services can be made

available on the screen, starting with a simple telephone-dialing keypad, but none of the applications runs on the appliance itself. They rather run elsewhere on the network, and simply use the appliance for input and output. A current embodiment of the appliance is designed to resemble a telephone, but the techniques developed here are applicable to a wide range of remotely-managed displays, and in particular those which incorporate audio I/O facilities.

The Terminal Device Hardware

Figure 6 is an overview of the terminal device electronics used in this specific embodiment. In the figure, network interface (101), connects the device to a ?10Base-T Ethernet or a similar broadband, packet-switched network. Telephone handset (102) incorporates a speaker and a microphone both of which may be wired to the rest of the device, may be cordless or may be built into the main body of the device. A hook switch or other mechanism (103) is used to detect if the handset is in use. Loudspeaker (104) is used to generate ringing sounds. This is driven from an amplifier of sufficient power to produce an easily audible ringing sound. It could also serve to produce audio output for hands-free telephone operation or other purposes such as music output. Microphone (105) is used for audio input in hands-free operation when such operation is desired. The microphone should be positioned away from the loudspeaker so as to reduce the possibility of feedback and instability during a hands-free telephony conversation. An adaptive echo cancellation device (106) such as Crystal Semiconductor CS6422 can be used to support simultaneous use of items 104 and 105 without excessive feedback.

Audio codec (encoder/decoder) (107) converts audio signals from analogue to digital and from digital to analogue, and includes means to alter the input gain and output volume under software control. Additional amplification for microphone inputs and speaker outputs may also be required. For telephony purposes, the codec supports an 8kHz sampling rate, should be monophonic and should have full duplex operation; it may use 16-bit precision to encode each sample. For other applications, such as music output and stereophonic operation, higher sampling rates may be desirable.

Audio path switching mechanism (108) selects which speaker or speakers are used, which microphones or microphones are used, and brings the echo canceller into or out of operation. This may be built into some models of codec or be implemented using CMOS (Complementary Metal Oxide Semiconductor) analogue switches.

A backlit LCD (Liquid Crystal Display), (109), or some similar display device having an array of individually addressable pixels, is used for visual display. In this implementation we use a 640*480 pixel TFT (Thin Film Transistor) LCD (mounted sideways to give a portrait appearance) with a bit depth of 16 bits per pixel (5 bits for red component, 6 for green, 5 for blue).

Touch screen input device (110), which can be operated using a stylus or finger, is located over the display, and has the same or a comparable resolution to the display.

Optional watchdog reset timer (111) resets all the terminal electronics one minute after being set by the software, unless it is disabled or set for a further period in the interim.

StrongARM SA1100 or an other appropriate microprocessor controls all the above hardware and executes the Viewer Software, in conjunction with whatever auxiliary electronics are required to enable the other hardware components to communicate with the microprocessor - some or all of which hardware may be built into the microprocessor. The processor should be of a type with sufficient processing power to drive the network, audio and display hardware reliably whilst decoding audio, video and graphics protocols at an acceptable rate; low power consumption may also be desirable. Alternatively some or all of the Viewer functionality may be implemented using dedicated hardware rather than software, in which case a simpler processor, perhaps embodied in a FPGA (Field Programmable Gate Array) may suffice as the device controller.

RAM (Random Access Memory) (113) holds temporary information used in the operation of the sofware on the microprocessor. Part of the RAM may also be used as a frame buffer to refresh the display.

ROM (Read Only Memory) or other nonvolatile memory (114) holding all operating system and viewer software which runs on the device. It or something in the device should also store a unique device identification number and (if using Ethernet) the Ethernet MAC (Medium Access Control) address (these may be the same).

Optional audio output connector (115) provides mono or stereo audio output to external equipment. Optional audio input connector (116) provides mono or stereo audio input from external equipment.

An optional serial port (127) may be useful for the initial installation or debugging of viewer software, or to connect to external digital equipment such as a keyboard or a printer.

Operating System

The device contains all the necessary software in ROM (114) to support concurrent use of all hardware components; to control software execution according to the demands of the hardware or the availability of data; to meet soft real time constraints on the timings of network transmission, sound playback and video display; and to implement the TCP/IP (Transmision Control Protocol / Internet Protocol) standards suite for communications on the network. This part of the device software is referred to as the Device Operating System. This specific embodiment runs a StrongARM version of Linux as its operating system.

Boot Procedure and Lifecycle

On powering up, or after being reset, the following steps are performed:

1. The Linux kernel and an initial file system image are retrieved from the ROM.

2. On entering the Linux kernel, a 60 second watchdog timer is started.
3. Networking and the TCP/IP protocol stack are configured using the DHCP (Dynamic Host Configuration Protocol) standard, or by some similar broadcast protocol for the discovery of network addresses and services. This may be achieved using a standard Linux DHCP client. The information returned may include the device's IP address, a netmask, a domain name and the addresses of one or more DNS (Domain Name Service) servers.
4. If the network interface cannot be intialised, there is no connectivity to a DHCP server, or the device is declined by the DHCP server, the device waits and then retries the above step. It may simply wait until reset by the watchdog timer. A warning message may be displayed on the screen.
5. If DHCP succeeds, the watchdog timer is set for a further 60 seconds.
6. The device then repeatedly executes the Viewer Software resident in its ROM. After executing the Viewer Software a given number of times (such as 16), the device resets itself and repeats the DHCP discovery step. This is to ensure that it remains correctly configured for the network to which it is connected, even if the configuration of the network changes.

- The following resources are available to the viewer software in this specific embodiment: Linux runtime environment; DNS lookup utility program; Calls to restart the watchdog timer or immediately reset the device; Memory mapped frame buffer (480 * 640 * 16-bit);Audio interface resembling OSS/Free; Control of audio path switching hardware; Control of echo canceller parameters; Sockets (TCP, UDP, Pipes) and TCP/IP stack; Poll for touch screen status changes; read touch screen status and coordinates; Poll for hook switch status changes; read hook switch status; Control of LCD backlight brightness may be provided; Real time clock (for intervals; absolute wallclock time need not be available).

Viewer Software

The Viewer Software consists of three parts which execute concurrently: an audio transceiver part, a graphics viewer part and a video receiver part. These may be implemented as separate threads or processes. In this specific embodiment the graphics viewer and video receiver share the same process and thread of execution but the audio transceiver is separate (Although here implemented entirely in software, some or all of the viewer functionality may be implemented in dedicated hardware).

Graphics Viewer

The graphics viewer listens on a well known TCP port (such as port number 5678). Once it is able to receive connections on this port, the viewer uses a DNS lookup (with a well known name such as bpserver appended to the domain name it received from the DHCP server) or some other IP based discovery mechanism, to find the IP address of its Server. The viewer then connects to the Server at a well known TCP port (such as port number 27406) and transmits a Registration message as described in the Protocols section. The viewer waits up to 30 seconds (or a sufficient time that the server will under normal conditions have been able to respond to the registration message and initiate a connection) to accept an incoming TCP connection on its listening port.

As in the operation of a VNC viewer (taking into account those protocol differences between Broadband Phone Protocol and VNC which are detailed in the Protocols section), if a connection can be accepted, the viewer and the server communicate with one another over this connection using the Broadband Phone Protocol as described in the Protocols section. Part of the Broadband Phone Protocol consists of audio and video control commands, which the Graphics Viewer relays to the audio and video parts of the Viewer Software. The Server uses the Broadband Phone Protocol to describe graphics which are to be drawn on the screen of the viewer. The Graphics Viewer draws these graphics on the screen, either directly to the hardware or its dedicated frame buffer or (as here implemented) via a temporary buffer in RAM. The advantage of using a temporary buffer is that entire updates can be made to appear on the

screen once they have been completely received and processed, rather than as each rectangle is received. The Graphics Viewer concurrently polls the hook switch and touch screen for activity, and transmits to the Server indications of any change in touch screen status or coordinates or of hook switch status, using the Broadband Phone Protocol. Correct initialisation and normal operation of the Broadband Phone Protocol results in the watchdog timer being set for a further period. The viewer may send or arrange to receive a small protocol message (such as to send a repeat of the previous hook switch message) every few seconds to test the validity of the connection and keep setting the watchdog timer.

All the Viewer Software terminates when the graphics viewer detects that the Server has closed the Broadband Phone Protocol connection, or when it detects a protocol violation on that connection.

Audio Transceiver

The audio transceiver is the part of the Viewer Software which transmits and receives audio to and from the network. Figure 7 shows its structure in terms of the major functional blocks.

UDP (User Datagram Protocol) datagrams are received from the network on some well known port (such as port number 5004) and are interpreted as packets carrying RTP (Real Time Protocol). They are matched against a list of patterns specified by the server, based on their originating IP address and port number (item 117), and discarded or assigned to one of several channels (here we support up to 4 channels). Packets from the same address might be distinguished by means of their RTP SSRC (Synchronization Source identifier), so that only packets from a single SSRC are assigned to one channel during any short period of time.

Incoming packets assigned to a channel are then processed to remove RTP headers and decode the RTP Payload (that is, the specific encoding used to represent the audio stream in digital form) into a sequence of digital samples (118). The specific embodiment can decode

payload types 0, 5 and 8 which correspond to G.711 mu-law, DVI4 and G.711 A-law respectively. Other payload types might be supported, for instance, for higher quality audio.

Samples of audio for each channel are collected in a FIFO (First In First Out) buffer (119) which has the ability to delete or insert synthetic samples to account for differences in the rate at which samples are provided and consumed, and to control the number of samples buffered so that any short term 'jitter' or mismatch between the rate of sample production and consumption tends to just avoid emptying the buffer. An empty buffer should yield silence when called upon to produce samples. The specific embodiment has four such buffers, implemented as ring buffers, and able to hold up to a maximum of 4096 samples in each. The buffer may, in conjunction with the RTP decoding procedures, provide some provision for the reordering of packets received out-of-sequence, and for synthesizing samples to conceal a failure to receive one or more packets, based upon analysis of RTP sequence numbers or timestamps on the packets received.

Samples from each receive buffer are mixed (120), together with the output of a local tone generator (121) which provides ringing sounds and other tones under the control of the Server. (In the specific embodiment the tone generator produces triangular waves at a frequency and amplitude specified by the server and can mix or alternate between two different tones to yield a dial tone or a warbling sound). The samples are sent to the audio codec device (Figure 6 107) for output to one or more of the loudspeakers.

The rate at which samples are written and read is determined by a clock on the audio codec. This timing information is conveyed to the transceiver by the audio driver in the device operating system (122).

Samples received from the audio codec are collected in a buffer until a given number of samples have been collected: these will form a single packet transmitted onto the network (123). In the specific embodiment, groups of 128 samples form each packet - because samples arrive at a constant rate of 8kHz, a packet of 128 samples will be available every 16ms. Optionally, the

tone generator (121) or a second tone generator may superimpose some sound on the outgoing samples, for instance to produce outgoing DTMF (Dual Tone Multi Frequency) tones.

Outgoing samples are encoded using one of the payload encodings permitted for RTP, and placed in an RTP packet (124).

Outgoing packets are transmitted on the network to a single (unicast or multicast) IP address and port number or transmitted multiple times to a number of IP addresses and port numbers (125).

The behaviour of all modules are controlled (126) by the Server, by means of Broadband Phone Protocol messages to the Graphics Viewer and interprocess communication from the Graphics Viewer to the Audio Transceiver.

It should be noted that blocks 117 and 118 are driven by the availability of packets from the network; blocks 120, 121, 122, 123, 124 and 125 are driven by timing demands of the audio codec; blocks labelled 119 have samples added by network activity and consumed by audio codec activity; and block 126 is driven by commands conveyed to it via the Graphics Viewer through an interprocess communication mechanism. The specific embodiment uses the Linux select() interface to meet all these demands within a single thread of execution.

It should be noted that the audio transceiver as implemented here does not participate in any end-to-end signalling for telephony, nor does it negotiate the payload format for packets or how they are to be routed. These functions are performed by the Server. Thus the audio packets may be routed directly from one Broadband Phone to another through the packet switched network; or they may be sent via a gateway; or they may be sent via the Server. Multi-party calls may be implemented using multicast, multiple unicast, or may be mixed, distributed or forwarded by equipment elsewhere in the network. The device supports any and all of these modes and makes no distinction between them.

An audio transceiver may optionally maintain statistics about the packets received for each channel and have some means to convey them to the Server to provide some feedback of network performance. The transceiver may optionally implement the RTCP (Real Time Control Protocol) standard or some subset thereof, to send and receive such statistical information to or from a remote device or, in conjunction with the Video receiver, to support synchronisation of audio and video presentation.

Video Receiver

A video receiver is not strictly required for a broadband phone, but is a useful enhancement. Although moving images can be conveyed to the graphics viewer using Broadband Phone Protocol, it may be convenient and more efficient to convey video to the device by means of a separate stream of UDP datagrams. This enables the device to display video which does not originate from the server, or which does not require reliable transport, or which could benefit from the reduced latency and overheads of UDP as opposed to TCP transport. The video receiver receives UDP datagrams from the network on a particular port, filters them according to their originating IP address and port number as directed by the Server, discarding datagrams from an unrecognised source. It interprets received datagrams as a stream of video frames, for instance, MPEG-1 (Motion Picture Expert Group) video video encoded using RTP.

A video receiver able to decode only MPEG-1 'I' pictures would be able to decode and display individual frames as soon as they have been received, without the need for additional buffering or frame reordering. If the transport protocol requires video frames to be divided into multiple datagrams, incoming datagrams would need to be buffered until each video frame became available. As implemented here, the Video Receiver displays video frames on the frame buffer, over the top of any graphics produced by the Graphics Viewer. The device as implemented here does not *transmit* video in any form. That could be achieved using separate equipment connected to the network, under control of the Server.

The Protocols

Ports and Protocols

Below the various protocols are summarised, with the ports on which they are used.

Broadband Phone Protocol

The viewer listens on **TCP port 5678** and accepts at most one connection at a time. This connection is Broadband Phone Protocol, which extends a modified RFB (Remote Framebuffer Protocol) (RFB4.3) which in turn is based on RFB3.3 as used in VNC. It closes the connection as soon as it detects a protocol violation.

Audio RTP

Received on **UDP port 5004** (assigned to RTP flows by IANA (Internet Assigned Numbers Authority)), and transmitted by a UDP socket bound to the same port number. In the RTP standard the implementation can receive and transmit G.711 ulaw and Alaw formats, and a simple ADPCM (Adaptive Differential Pulse Code Modulation) compression format "DVI4" as specified in the RTP basic audio/video profile.

Audio RTCP

RTCP (Real Time Control Protocol) is an end-to-end protocol for exchanging information about network conditions, and for audio/video synchronisation. It is not used in the specific embodiment, but may be required for interoperation with some other endpoint equipment. It might be implemented by the device using UDP port number 5005.

Video RTP

A video format such as MPEG-1 Video embedded in RTP may be received on **UDP port 5006** and displayed on the screen, overwriting parts of the RFB display. MPEG-1 Video format is described in.

Video RTCP

RTCP for video flows is not implemented in the specific embodiment, but might be implemented by the device using UDP port number 5007.

Device Registration Protocol

The viewer connects out to a specified port (such **TCP port 27406**) on the Server to indicate that it has started listening for an incoming Broadband Phone Protocol connection. It discovers the Server's IP address by performing a DNS lookup or using some other broadcast or DHCP based discovery scheme.

The registration message comprises fields describing the device's unique hardware identification code, which is encoded at the time of manufacture in nonvolatile storage within the device and in the specific embodiment is the same as its Ethernet MAC (Medium Access Control) identifier; the IP address of the device as conveyed to it by the DHCP server; and the port number on which it can accept a Broadband Phone Protocol connection.

Broadband Phone Protocol

RFB4.3

The Broadband Phone Protocol is an extension of Remote Framebuffer Protocol RFB4.3, which differs from RFB3.3 used by VNC in the following ways:

1. There are an unspecified number of rectangles in an update. The field formerly nRects is now ignored. Each update terminates with: CARD16 dontcare; CARD16 dontcare; CARD16 0; CARD16 0; CARD32 dontcare, i.e. a pseudo-rectangle (a protocol element which, syntactically, can be transmitted whenever a rectangle could have been transmitted) having zero width and zero height (where CARD16, CARD32 are as defined for VNC).

2. **Offset** pseudo-rectangle, a rectangle header having nonzero but meaningless width and height fields, offsets future rectangles by its x,y coordinates (modulo 2^16), including both the source and destination rects of a CopyRect. Multiple offsets are cumulative. The offset is reset to (0,0) at the start of each update. Rectangle type 6.

3. **CopyRect** is required to copy correctly pixels that have just been drawn by earlier rectangles of the current update, including the results of a prior CopyRect. (in VNC this was not attempted).

4. New **TransRRE** encoding type, like RRE but with a transparent background (no background colour is transmitted). Subrects must still lie within the bounding rectangle. Rectangle type 3.

5. Transparent extension to **HexTile** encoding. An extra tile flag (32) now signifies "transparent", i.e. that the background for this tile should not be drawn. This flag cannot be used on raw tiles. The maximum number of subrects in a tile is 255 (in VNC it was expressible but never useful to have this many).

6. **JPEG** encoding type, consisting of a CARD32 length field followed by that number of bytes which encode a single Baseline JPEG (Joint Photographic Expert Group) image with Y'Cb'Cr' components as in JFIF (JPEG File Interchange Format specification. The width and height given in the SOF_0 (Start of Frame type 0) marker must match those of the rectangle header. Rectangle type 7.

7. Extension **server messages** having message types between 64 and 127. Each such message type byte is followed by a single padding byte and a CARD16 which contains the number of bytes to follow. These are intended to carry additional types of message not defined in RFB.

8. **Hook event** client message, having message type 8, followed by a single byte, the meaning of which is not defined in RFB (under the Broadband Phone Protocol, zero is on-hook and nonzero is off-hook)

9. Optional **extension client messages**, like extension server messages having types 64-127 and a 16-bit length field. No such messages are defined here.

Broadband Phone Protocol extensions to RFB4.3

Broadband Phone Protocol extends RFB4.3 by implementing a number of Extension Server Messages. These messages are used to control the reception of video and reception and transmission of audio by the device. Audio control commands include the following:

- Set which microphone(s) or loudspeakers(s) are in use, and whether the echo canceller is in use.
- Set the output volume.
- Set the input gain.
- Set the payload type used to encode outgoing packets, and whether or not to suppress transmission of packets containing only silence.
- Stop all transmission and reception of packets.
- Set the address and port number to which to transmit packets for one of a number of simultaneous destinations.
- Stop transmitting packets to a particular destination.
- Set the predicate (in terms of originating IP number and port number) for accepting packets and assigning them to a particular reception channel.
- Stop accepting packets on a particular reception channel.
- Start generating a tone of the specified frequency and volume. If multiple tones are supported, generate one or more tones with the specified frequencies and volumes simultaneously or in alternation at a given frequency.
- Stop generating tones.

Video control commands include the following:

- Set the predicate (in terms of originating IP number and port number) for accepting packets of video. Currently only one incoming flow of video can be processed by the device at one time.
- Stop accepting video from any source.
- Set the position on the screen at which video is to appear.

Note that all of these commands are idempotent and can be repeated if for any reason they are 'forgotten' by the device.

Broadband Phone Protocol: optional additions

Some other ways in which RFB4.3 might be extended to implement a broadband phone protocol include:

- **Colourmapped rectangles** or updates, that is: graphics temporarily drawn from a restricted subset of the colours available on the display, colours which are separately indicated by the Server by means of some other message or pseudo-rectangle. This might be used for the encoding of glyphs or other graphical units which need to appear repeatedly but in different colour schemes.
- **Packetisation** of messages or rectangles in which each message or rectangle could contain at its head a field encoding its length, or be transmitted as a series of fragments (whose headers encode their lengths) followed by a trailing end marker.

Server

Creation and management of server sessions

When a device is connected to the network, a number of different entities on the network are involved in ensuring the device is served appropriately. The following description assumes that the device is connected to an Internet Protocol (IP) network, though the principles apply to any similar network. (In this document, the term *server* broadly means all the hardware and software on the network needed to serve a broadband phone device.)

The device initially must obtain an appropriate IP address for itself. This is usually done using Dynamic Host Configuration Protocol (DHCP), as described in the "device" section. Standard DHCP servers can be run on the network for this purpose. The next step is for the device to find the IP address(es) and port(s) of the broadband phone *factory* service. This may be done through DHCP options, looking up a well-known name in the Domain Name Service (DNS) or other resource location service, or falling back to hardcoded defaults.

The device then makes a connection to the factory service, giving it relevant information about the device, for example in the form of name-value pairs. (The factory service is a software tool that provides particular services, in this case starting or locating a suitable session for the device.) This information includes the device identifier (currently an ethernet MAC address), as well as the IP address of the device (it may also include information on which protocols the device supports, etc). If a given factory is unavailable, there may be other factories available on the network that the device can try.

Having successfully received a connection from a device, the factory starts a *session*, or locates a suitable existing session, on an appropriate machine or machines. The session is the software entity that provides the graphics for the device's screen, interprets the pointer (or other input) events it sends back, and manages any connections (audio & visual) to other devices. The session may be a single (unix-style) process on a particular server machine, or it may be distributed amongst several processes, possibly on multiple server machines. The session has one or more network connections to the device, which may be initiated from either the device or the session as appropriate. Some means of authentication may be required by both ends on these connections, to ensure that both the device and the server can trust each other (for example using public/private keys). The session may also provide a login prompt on the device's screen to make sure that the person using the device is authorised to do so.

The factory and sessions depend on the *station directory* for information about devices and *stations*. A *station* can be thought of as a "logical phone", which has an identifier, akin to a "phone number", and other useful information, such as a textual name. Each physical phone device known to the system is associated with a given station in the station directory.

When the factory receives a connection from a particular device, it looks up that device in the station directory to see what its associated station is, and decides what kind of session it should start (or locate). Different kinds of session may be started by the factory depending on a

number of criteria, including the information passed by the device to the factory, and information stored in the station directory. For example, for a device which is not known to the factory, the factory may start a minimal session which simply displays an error message on the screen. Sessions for particular stations will be configured accordingly - for example those for a particular individual may be configurable and personalised to them, and may require the user to login before being able to use it.

In addition, the factory may decide to start sessions on a range of server machines according to certain criteria. This may be to balance the load evenly between a number of machines, or the factory may choose a server machine which is somehow "close" to the device or other servers to avoid unnecessary network load.

There are a number of administrative tools for managing sessions and stations. Tools are provided for creating and deleting stations, as well as altering the information stored about them in the station directory, including associating stations with devices. Sessions can be killed or replaced with different sessions for particular stations and devices.

Thin-client graphics

The session is composed of one or more processes which, amongst other things, provide the graphics for the device's screen and interpret pointer (or other input) events. The "Broadband Phone Protocol", described in the "protocols" section, is used to communicate with the device for this purpose.

There are broadly two ways of generating the graphical part of the Broadband Phone Protocol. The first is to use a similar technique to existing VNC (Virtual Network Computing) servers. The session includes an area of memory (framebuffer), which represents what should be displayed on the device's screen. Applications render into the framebuffer by some means. For example, the framebuffer could be part of an X VNC server, and the applications X clients, which

render into the framebuffer by sending X protocol requests to the X VNC server. Alternatively the framebuffer could be on a PC running MS Windows, and the applications Windows programs which use the Windows graphics API to render into the framebuffer. In principle any graphics system can be used to generate the pixels in the framebuffer. This is a powerful technique for accessing applications written to use an existing graphical system such as X.

The session updates the device's screen by simply sending rectangles of pixels from its framebuffer, encoded in some form. It can do so intelligently by knowing which areas of the framebuffer have been altered by applications, and only sending those parts of the framebuffer to the device. Input events from the device are fed back to the session's applications. More details about this are described in published VNC documentation.

An alternative way of generating the graphical part of the Broadband Phone Protocol is to write applications which generate the protocol directly without use of a complete framebuffer on the server side. This is best done by use of a toolkit, which provides higher-level concepts to the application such as buttons and fonts. The advantage of this approach is that it can be more efficient in terms of both memory and processing on the server. In practice a combination of the two techniques is used.

In addition to graphics, the session also controls other aspects of the device. Again the Broadband Phone Protocol is usually used for this purpose. Such controls include setting various audio parameters, tone generation and setting up of audio connections to other devices. See the "protocols" section for more details.

Audio and Video Presentation

The Server may use the Broadband Phone Protocol to instruct the device to accept audio or video streams from the server or from a streaming media server and may itself transmit RTP

packets or cause the media server to transmit packets to the device. In this way, the server may display videos or cause sounds to be played on the device.

Signalling

As described in the "device" & "protocols" section, the devices use the RTP protocol for sending and receiving audio and other media streams. However, setting up these streams requires "signalling" between the participants in a call. In the case of the broadband phone system, the entity responsible for signalling is the session, rather than the device itself. Between broadband phone sessions, we use our own signalling system built on top of the CORBA distributed object framework. This allows us to add extra features to calls such as shared graphical applications.

For interoperating with other IP telephony devices or the PSTN, standard signalling systems must be used, the most common of which are SIP and H.323. By talking one of these protocols to a suitable gateway to the PSTN, a broadband phone session can make and receive ordinary voice telephone calls on behalf of a broadband phone device, to and from which the audio stream can be routed.

Applications and Services

An application or service user-interface is activated by touching a graphical icon representing it, or selecting it from a list on the screen, or by recognising a spoken word or words, or by other means, similar or different.

Phone Dialer

The phone dialer presents a user-interface which allows the user to communicate with another phone, either broadband phone or conventional phone. The user-interface consists of a number of screen-buttons representing symbols, including but not limited to the digits 0-9 * and #, and functions. When a screen-button representing a symbol is pressed, the symbol it represents

is concatenated to the end of a sequence of symbols which is displayed on the screen. The sequence of symbols can be edited by first selecting a symbol or symbols, and then pressing a screen-button representing the delete function. Another screen-button representing the clear function allows the sequence of symbols to be cleared. The call is attempted when either the screen-button representing the dial function is pressed, or the sequence of symbols represents a valid phone identifier as determined by the signalling system. The call is attempted by passing the phone identifier to the signalling system, which is part of the session on the server. Screen-buttons for other common phone functions are provided, including redial, memory, and pickup.

Information about the party being called, such as a graphical image of the party, or details of a company, or a message for any incoming caller, or a specific incoming caller can be displayed on the caller's screen. When a call is being attempted, a screen-button representing the hangup function can be pressed to indicate to the signalling system that the call should be aborted.

When a call has been accepted, the hands-free microphone and speaker are connected to the audio output and input devices respectively of the other party. A screen-button representing the mute function allows the connections to be temporarily broken, until the screen-button representing the un-mute function is pressed. Other screen-buttons for common phone functions including hang-up, conference, put on hold and redirect are displayed.

When a call is in progress, the user can interact with other applications and services whilst speaking with other parties, including a notepad, calendar or calculator, by selecting the application or service in a similar manner to that described above. A mechanism such as a graphical icon to return to the phone dialler user-interface can be provided as a shortcut.

When a call is in progress, certain applications and services can be shared with the other parties. An application or service can be selected to be shared in a similar manner to that

described above. Examples of applications and services which can be shared is given below. A mechanism such as a graphical icon to return to the phone dialler user-interface can be provided as a shortcut.

The handset microphone and speaker can be used as alternative to the hands-free microphone and speaker by picking up the handset thereby releasing the off-hook switch. A screen-button to switch back to hands-free microphone and speaker is then displayed.

The audio channel from other parties can be analysed or interpreted and the result shown simultaneously on the receivers screen, for example a lie detector. This would require the audio stream to be sent additionally to a process on a server, which would require an addition signalling message to be sent to device originating the audio stream.

Incoming Call

An incoming call is alerted by the phone making a ringing sound, or a graphical message on the screen or both. Information about the calling party can be shown on the screen, such as a graphical image of the calling party, or graphical details of a company, or a graphical message for the person answering the call. Screen-buttons representing accept and reject functions are displayed.

When a call has been accepted, the hands-free microphone and speaker are connected to the audio output and input devices respectively of the calling party. A screen-button representing the mute function allows the connections to be temporarily broken, until the screen-button representing the un-mute function is pressed. Other screen-buttons for common phone functions including hangup, conference, put on hold and redirect are displayed.

When a call is in progress, the user can interact with other applications and services whilst speaking with other parties, including a notepad, calendar or calculator, by selecting the

application or service in a similar manner to that described above. A mechanism such as a graphical icon to return to the Incoming call user-interface can be provided as a shortcut.

When a call is in progress, certain applications and services can be shared with the other parties. An application or service can be selected to be shared in a similar manner to that described above. Examples of applications and services which can be shared is given below. A mechanism such as a graphical icon to return to the Incoming call user-interface can be provided as a shortcut.

The handset microphone and speaker can be used as alternative to the hands-free microphone and speaker by picking up the handset thereby releasing the off-hook switch. A screen-button to switch back to hands-free microphone and speaker is then displayed. An incoming call can be accepted directly by picking up the handset.

Directory Services

Directory listings of names or images are displayed on the screen. By touching a name or image, the phone identifier associated with that name or image is dialled directly (by interaction with the server, as above) and automatically. Office directories and staff-lists can be displayed in this way, and the service can use existing databases such as LDAP. Residential numbers and yellow pages can be similarly displayed. The information available with these services is accurate and up to date. The physical location of the phone is known because of the physical location of the network connection it is attached to is known, and so geographically local information directory services can be provided. These can be ordered by distance, to find the nearest matching directory entry. Directories can be organised into maps, including an office layout or town street map, allowing a location or facility to be dialled directly by touching that part of the map. Local maps can be centred around the location of the broadband phone. Personal directories of names or images and phone identifiers can be created.

Calculator

Screen-buttons representing a numeric keypad and the functions normally found on electronic calculators allow a calculator to be implemented. An area of the screen is used to display the numbers entered and calculated.

Notepad

The notepad provides an area of the screen containing a background image, including but not limited to a plain white image, on which the movements of a pen or finger touching the screen are reflected by drawing a series of graphical objects such as a line between successive pen or finger positions. Properties of the graphical object such as size, shape, texture, colour can be varied by selecting from a menu provided by the notepad application. The note can be edited by selecting a pen of the same colour as an element of the background, which can effectively be used as an eraser. A note so created is automatically and periodically saved on the server, provided it has been changed since the last time it was saved.

The notepad allows several notes to be created and exist simultaneously. A screen-button representing the new function creates a new note. Screen buttons representing backwards and forwards allow the user to select which note is shown on the screen. A screen-button representing delete allows a note to be deleted. The number of the current note, and the total number of notes is displayed in an area of the screen.

A note can be sent as email in the form of a graphical attachment. The email address to which the note is sent is entered by screen-buttons organised to simulate a computer keyboard, or by handwriting recognition, or voice recognition or other means. Alternatively, the email address can be selected from a list of addresses which have been used previously, which may be ordered with most-recently-used first.

A note can be sent to a networked printer. The printer can be chosen from a list, or the name or address of the printer can be entered in the manner of an email address as described above. A note can be sent as a message flash directly to the screen of a set of other broadband phones. A message flash displayed on another phone will automatically disappear after a time, or earlier if explicitly dismissed by the recipient touching the dismiss screen-button.

The notepad can be shared with the other parties in a phone call. All parties can simultaneously see and interact with a representation of the same note.

Piano

The piano application consists of a number of screen buttons arranged to look like, for example, a two octave piano keyboard with white and black keys. While a key is pressed, the colour of the screen button is changed and the corresponding note is output on the hands-free speaker, or handset speaker if the handset is lifted. There is a screen-button to change the pitch of the notes output by one or more octaves, and screen-buttons to add various acoustic effects, including volume, sustain, change of timbre. There are screen-buttons to record a sequence of notes and playback the recorded sequence. The piano can be shared with other parties in a phone call. All parties can simultaneously see and interact with a representation of the same keyboard. The keys pressed by each party are shown in a different colour, and the notes sound simultaneously allowing duets etc to be played.

Chess

The chess application comprises a graphical representation of a chess-board, with chess pieces. The chess pieces are screen-buttons, which can be moved to another square on the chess-board. If another piece already occupies that square it is taken and replaced by the piece moved. There are screen-buttons to reset the board to the normal chess starting position, and to undo moves back to the previous normal chess starting position. There is a screen-button which turns on checking for legal chess moves, and a screen-button which lets the computer play one

side of the chess game. The chess application can be shared with other parties in a phone call. All parties can simultaneously see and interact with a representation of the same chess-board. Other games can be provided similarly, such as cards. A card playing application would deal cards to the parties. Each party sees only their remaining cards, plus the cards which have been played.

Minesweeper

The minesweeper application is a version of the popular game to deduce where the hidden mines are. The board consists of a number of unmarked screen-buttons squares. To uncover a square, simply touch the screen-button. If it is a mine, the game is lost. If a number is revealed, it says how many mines are in the adjacent squares. If it is deduced that a square is a mine, it can be flagged by gesturing with a stroke beginning in that square. If a square is incorrectly flagged it can be un-flagged by making another stroke starting in that square. The number of bombs remaining is indicated on the screen. The game is won if the location of all mines is correctly marked. Sound effects are played on the hands-free speaker, or handset speaker as appropriate, including sounds for revealing a number, revealing a bomb and winning the game.

Album

The album application allows a collection of images to be browsed. Pages of thumbnails are shown. The user can zoom-in by making a closed stroke, such as a circle, around a set of thumbnails. Zooming-in causes a page of thumbnails to be shown which were contained within the closed stroke, and which are scaled to fit the page. Ultimately a single image is show at maximum possible size for the page.

A screen button allows the user to zoom back out, and switch between pages. Images can be printed on a networked printer. The images can be entered into the album through a network connection to a digital camera, including a wireless connection. The album application

can be shared with other parties in a phone call. All parties can simultaneously see an image chosen from the album. Images can be categorised or organised either manually or automatically into pages of thumbnails to facilitate searching and browsing and to find similar images.

Video

Video from JPEG or MPEG IP cameras attached to the network can be viewed. One application is for security purposes. By positioning cameras close to, or even integrated with the phone a video phone can be constructed, in which the parties in the call can see video from the other parties. Alternatively, archives of video can be browsed and viewed. These may have an accompanying sound track. Video archives can be categorised or organised either manually or automatically to facilitate searching and browsing and to find similar video clips.

Music

An online catalogue of digitally represented music or spoken word can be browsed. Albums or individual tracks can be selected and played. Tracks can be categorised or organised either manually or automatically to facilitate searching and browsing to find similar tracks.

Calendar

The calendar application shows days of the month. The month and year can be chosen by forward and backward screen-buttons. Each day is a screen-button, which when pressed allows a note to be created similar to the notepad, for example to contain appointments. Days with notes are differentiated by colour, as is the current day. When a day with an attached note occurs, the note is automatically brought to the front of the screen once, so that the person will see it as a reminder. The application shows the time in the local timezone, together with the time in other world timezones.

Web

A web browser allows access to the internet. The touch-screen allows links to be followed. When text is required to be input, the pen or finger can be used to use a screen-keyboard as described above, or character recognition. Alternatively, voice recognition can be used to enter words, characters or actions.

Shopping and Reservations

When an online shopping or reservations number is dialled, or a shortcut icon is pressed, or a name selected from a list or some such, a portion of the screen is updated and provided by the 3rd party company. This can be a shopping catalogue, with screen-buttons to browse and select, or a menu of choices for an information or reservation line. The act of choosing or selecting from a catalogue or menu may result in an audio connection to an operator at the 3rd party company, who is able to see the same information that the caller sees.

Fax and Mail

Fax and mail can be received and displayed. Screen-buttons allow the items to be managed including delete and reply. A reply may be created as a graphical entity with pen or finger strokes, or as text entity with handwriting recognition, or speech recognition. A reply may be created by pen or finger strokes on top of the incoming item, and sent as a graphical item.

Other Applications

The system described, or one based on the same concepts, could also drive a range of other devices. Some variations from the desk-phone-like appliance currently used would include:

- Audio and graphics on a single device with no handset;

  The handset could be discarded to give a tablet- or PDA-like device with speakerphone capabilities. This could be a portable device using a wireless network.

- Audio and graphics on separate devices but still connected by a network.

  This variation might include a fixed display with a cordless headset, or wall-mounted display panels near the phone handset, or separate cordless graphics and audio devices.

There is also then no need to keep a one-to-one relationship between audio device and graphics device.

- Graphics alone

  The system can be used to drive networked display devices for which an audio connection is unimportant. (eg. airport flight information display boards, road traffic signs, car dashboards, controls for home automation/entertainment/heating/alarm systems).

- Audio alone

  A device driven by the server but only using the audio facilities of the system could provide a remote or extra audio connection.

- Multi-channel audio

  Multiple channels of audio could be sent to a device to provide stereo or surround-sound experiences. The different channels might also be used to provide audio in different languages for several users watching the same display.

- Multi-channel video

  More than one display could be driven, to provide a larger image spread over several displays, or to provide binocular vision for use with 3D glasses or head-mounted displays.

- 'Proxy' devices

  These would connect to the server as before, but use the graphical and/or audio facilities of another device for the actual input and/or output. An example would be a TV set-top box which would display on the TV, use a remote control for pointing, and a home hi-fi system for audio. One might also imagine a system using a projector and a laser pointer as an alternative to an LCD and touchscreen.

Combinations of the above variants are, of course, also possible. Finally, the device may not exist as a separate physical appliance at all. The thin-client software may be run on a conventional PC or workstation to provide a 'soft' phone on a more general platform.

Server

It is important to emphasise that the updating of one display need not originate from a single server machine or process. The server software and applications may themselves be distributed; portions of them may run on more than one machine even if only one is responsible for updating the display. This might be done for a variety of reasons including load-balancing, security, more efficient use of resources or simply easier management. More than one server might generate graphics for a given screen. A typical use would be an advertising banner at the top of the screen coming from one company while the main contents come from another. The separate areas might be sent to the device independently, or might be 'merged' by one overriding server. Lastly, a given display may be sent to more than one device. It might be desirable for all the phones in one house to appear to have the equivalent of 'the same number', for example, and this could apply to the graphics as well as the audio. Another scenario in which this is useful is the operator or 'helpdesk' being able to view and interact with the same screen display as the user with the query. The display may also be moved between devices (see below).

Input and Output

Many of the services available on the system might need text input, for example to enter an email address, or keywords for a search, or even to enter longer messages and documents. At present this is done by displaying a pop-up QWERTY-like keyboard on the screen, but alternatives include:

- Handwriting recognition services on the network, to which the pen strokes from the screen are sent. Users might even choose to subscribe to the service which most effectively recognises their writing, or one particularly customised for their language and character set. Additionally, the recognised text would not have to be in conventional handwriting, but could use a modified alphabet such as are currently used on many pen-based PDAs.
- Speech recognition could also be used, and the display of the text combined with pen-based interfaces could make for efficient correction of imperfectly-recognised text.

Text-to-Speech and Speech-to-Text

The combination of an audio device with a graphical display could be important for those suffering from speech difficulties or aural or visual impairments. Some examples include:

- Audio cues could accompany the display of information on the screen and could provide feedback when interacting with it. The text on a button could be spoken as a user's finger moves over it, for example, and a click sound emitted when the button is tapped.
- People with speech difficulties could write words on the display to augment or replace a spoken conversation. These could be transmitted as graphics, or converted into speech.
- Speech recognition systems could provide 'subtitled phone calls' for the hard of hearing, or translation services for those trying to follow a conversation in another language.

Interesting variations on the described applications include:

Dial-by-Map

Directory services have been described earlier as an alternative to dialling using more traditional telephone numbers. But a wide variety of other methods might be employed, including 'dialling' by clicking on a map. If the map were a floor plan of a building, this could be used to call a particular room. It it were a map of a larger area, it might call a particular class of service for that location - the police station covering that area, for example, or a bus company with stops in the vicinity.

Voice Menus

The rather tedious voice-based menus of the "for Sales, press 1" variety, could be more easily navigated if simultaneously presented on the screen. The graphics might be provide by the company whose menu was being navigated, or by a third party through the use of an agreed 'menu protocol' which would provide the textual menu to accompany the spoken one, or even by

a speech-to-text system recognising the spoken menu and transcribing it into a graphical equivalent.

Voice Mail

A variety of improvements to the user experience of traditional voicemail systems become possible with graphical support. One example might be an email inbox-style display of waiting messages. In addition to options for managing the messages, CD-player style controls could provide for playback, pausing, forward and rewind etc. of the audio. Speech-to-text systems might provide automatic 'subject' lines or search terms for the voice messages.

Customisation

The service provided on a device could be customised to an almost infinite degree, since every pixel on the device's display can be modified remotely. Since the system allows great flexibility in the source and routing of the pixels, the customization could be provided by many different parties. Customisation might be done, for example, based on the provider of the basic service, the third-party services to which they or the user have subscribed, the identity of the device, or the identity of the user (see below), to name just a few. Some customisations may be particular to the services provided, and others might be more generally applicable. A user suffering from red/green colour blindness, for example, might arrange for the display to be routed via a service which transformed certain shades of red and green into more easily distinguished colours.

Personalisation

If the device (or, more precisely, the server controlling the device) knows the identity of the user, the display and the services presented may be personalised for that user in any way. Moreover, if the user moves to a new device - a public call box, for example - and establishes their identity to that device, their personal settings, address books, configurations etc could follow them to the new device. The user's identity could be established in a number of ways: by 'logging

in' using prompts presented on the screen, by writing a signature on the screen, by swiping a card or presenting a similar 'key' to the device, or by biometric methods. In the simplest form, a particular phone device might identify the user ("This is Bob's phone, so the user must be Bob"). And the 'transfer' of the user's preferences to a new location could be done using some agreed protocol, or simply by asking the user's normal server to interact with a new device.

Network

Other underlying networks are possible. Wireless (local or larger-area), optical fibre, infrared, satellite or cable could be used for part or all of the communications, for example. Separate networks might be used for audio and graphics. We might imagine using a wireless network to give a cordless handset for use with wired displays, or choosing one style of network for the low-bitrate reasonably consistent traffic of the audio channel and another for the very bursty and asymmetric traffic of the graphical channel.

CLAIMS

1. A communication system comprising: a first endpoint device having an audio transducer and a display screen; a first server which has residing therein at least one application which affects the image on at least one portion of the screen and which server performs signalling for controlling an audio connection between the first endpoint device and a remote device; and a network connecting the first endpoint device to the server by a non-dedicated communication path.
2. A system as claimed in claim 1, in which the first server contains sufficient information to be able to regenerate an image on at least one portion of the screen.
3. A system as claimed in claim 1 or 2, in which the network is a packet switching network.
4. A system as claimed in any one of the preceding claims, in which the first endpoint device contains insufficient information to permit regeneration of the image on the at least one portion of the screen.
5. A system as claimed in any one of the preceding claims, comprising a plurality of second endpoint devices, each of which is of the same type as the first endpoint device.
6. A system as claimed in any one of the preceding claims, comprising a plurality of second servers, each of which is of the same type as the first server, the first and second servers being connected together by the network.
7. A system as claimed in any one of the preceding claims, in which the network includes a public switched telephone network.
8. A system as claimed in any one of the preceding claims, in which the first endpoint device comprises a frame buffer for storing display data in a display format ready for display by the screen.
9. A system as claimed in claim 8, in which the first endpoint device comprises an updating circuit for replacing data in the frame buffer with fresh data in a transmission format from the first server.

10. A system as claimed in claim 9, in which the first endpoint device comprises an interface for interfacing between, on a first side, the updating circuit and the transducer and, on a second side, the non-dedicated communication path.

11. A system as claimed in claim 10, in which the non-dedicated communication path is a single channel path carrying audio and non-audio data.

12. A system as claimed in any one of the preceding claims, in which the first endpoint device comprises a position measuring system for measuring the position of a pointer relative to the screen.

13. A system as claimed in claim 12 when dependent on claim 10 or 11, in which the position measuring system comprises a position measuring transducer and a converter connected to the interface on the first side for converting the measured relative position to data representing coordinates of the measured relative position.

14. A system as claimed in any one of the preceding claims, in which the at least one application supplies the data for affecting the image to the first endpoint device in response to a request from the first endpoint device.

15. A system as claimed in claim 9 or in any one of claims 10 to 14 when dependent on claim 9, in which at least one application converts the data for affecting the image from an application format to the transmission format.

16. A system as claimed in claim 14, in which the at least one application supplies data for affecting the image to the first endpoint device via a first in/first out buffer.

17. A system as claimed in claim 16, in which, when the buffer contains first and second items of the data for affecting the image, which first item was supplied to the buffer before the second item and which first and second items contain image data for the same region of the screen, the at least one application deletes the image data from the first item.

18. A system as claimed in claim 15, in which the at least one application forms the data for affecting the image as a sequence of blocks, each of which comprises a polygonal region of the screen and coordinates representing the position of the polygonal region on the screen.

19. A method of operating a communication system of the type comprising: a first endpoint device having an audio transducer and a display screen; a first server which has residing therein at least one application which affects the image on at least one portion of the screen; and a network connecting the first endpoint device to the server by a non-dedicated communication path, the method comprising performing, in the server, signaling for controlling an audio connection between the first endpoint device and a remote device.

20. A computer program for controlling a computer to perform a method as claimed in claim 19.

21. A storage medium containing a program as claimed in claim 20.

The Patent Office

45

INVESTOR IN PEOPLE

Application No: GB 0016661.1 **Examiner:** B.J.SPEAR
Claims searched: 1-21 **Date of search:** 5 March 2001

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.S): H4K (KFH,KOD3,KOD4,KOD8)

Int Cl (Ed.7): H04L 29/00;H04Q 11/04

Other: Online:WPI,EPODOC,JAPIO,INSPEC

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
XY	GB2332126A	(Samsung) Whole document, eg abstract, Figs. 1,2 and p 5 l 31-p 9 l 9.	1,2,5,6,19-21 at least
XY	GB2321732A	(IBM) Whole document, eg abstract, Figs 4 and 10, and p 5 l 35-p 13 l 33.	1-3, 7, 14,19-21 at least
XY	EP0903903A2	(Northern Telecom) Whole document, eg Figs.1,3; col 7 l 49-col. 11 l 18	1,2,5-7,19-21 at least
Y	EP0355697A2	(Hitachi) Whole document, eg p 3 l 38-p 4 l 47;p 14 l 38-p 15 l 2.	12
XY	WO97/42728A2	(Webline) Whole document, eg abstract, Fig. 1, and p 13 l 7-p 20 l 15, p 36 l 11-p 37 l 22.	1-5,7,19-21 at least
XY	Taniguchi K et al "Internet Video- on- Demand System Architecture-MINS"IEICE Trans. Commun. Vol. E79B- No. 8 Aug. 1996 Pages 1068-1075,especially Figs.1 and 2. ISSN 0916-8516		1-4,19-21 at least
XY	Libman R E at al "The Interactive Video Network :An Overview of the Video Manager and the V Protocol. AT&T Technical Journal Vol.74 No 5 September/October 1995, pages 92-105 ISSN 8756-2324		1-4.19-21 at least

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

This Page Blank (uspto)